SNS の投稿内容に含まれる地域情報を用いたアカウント到達可能性算出モデルの検討

吉國　綺乃†　　渡辺知恵美††　　小林　一郎†

† お茶の水女子大学大学院人間文化創成科学研究科　〒 112-0012 東京都文京区大塚 2 丁目 1 － 1
†† 筑波大学システム情報系情報工学域　〒 305-8577　茨城県つくば市天王台 1-1-1
E-mail: †{yoshikuni.ayano,koba}@is.ocha.ac.jp, ††chiemi@cs.tsukuba.ac.jp

あらまし　近年，ソーシャルネットワーキングサービス（SNS）の利用において自身が投稿した内容やプロファイルがどの程度のプライバシリスクになっているかを把握する必要性が増している．我々は SNS におけるプライバシリスクの提示指標として，アカウント到達可能性を定義している．アカウント到達可能性は，攻撃者が利用者の既知のアカウントから別のアカウントを見つけ出す可能性を表す．本論文ではアカウント到達可能性を求める具体的な手法のひとつとして，SNS の投稿内容に含まれる地域情報をもとにアカウント到達可能性算出モデルを検討する．

キーワード　ソーシャルネットワーキングサービス，プライバシ，Web

1. 背　　景

近年ソーシャルネットワーキングサービス (以下 SNS とする) の普及により，世界中で利用者が増えている．主流である Facebook(注1) のアクティブユーザは 11 億人を超え，日本国内だけでも 2013 年 6 月現在で 2200 万人を超えている [7] ．また主流 SNS の一つである Twitter(注2) のアクティブユーザは月間 2 億 1800 万人を超えている [8] ．利用者は友人や知人，同じ趣味を持つ他者とのコミュニケーション，またオンラインのゲームなどをするために SNS を利用している．SNS ではプロフィールの公開，日記やショートメッセージの投稿，またチャットのやり取りなど，さまざまな方法でコミュニケーションをとることができる．しかしながら一方では SNS でのトラブルが原因となり，利用者の個人情報が取得されるという事例が発生している．炎上事件が例に挙げられる．きっかけはさまざまであり，いつ，だれが，どのように被害にあうかはわからない．SNS を利用している人は誰でも被害者になり得るのである．サイバーストーカーと呼ばれる攻撃者は，利用者が利用している SNS の情報をもとに個人情報を多く取得しようとする．これらの個人情報は自身が気が付かないうちに，投稿内容やプロファイルに自身で公開していることが多い．サイバーストーカーは利用者が公開している情報の組み合わせで，多くの情報を取得していく．

SNS の利用者にとって取得されたくない情報の一つに，居住地などの個人に関する地域情報があげられる．Twitter のつぶやきだけで個人が特定できるか調査した報告 [9] では，全く知らない Twitter の利用者をその利用者の Tweet だけで本人を見つけ出すことを実際に行っている．彼らはまず地域名で検索した結果上位に挙がってきた Twitter ユーザを対象ユーザとし， Tweet を追跡することでリアルタイムで対象ユーザを見つけ出している．実際に見つけられた利用者は，全く知らない人に個人が特定されてしまうと考えてはおらず．Twitter の利用者がいかに安易に地域情報を公開しているかがわかる．

このように SNS の利用において個人情報が取得することを防ぐためにも，自身が投稿した内容やプロファイルがどの程度のプライバシリスクになっているかを把握する必要性が増している．我々は SNS におけるプライバシリスクの提示指標として，アカウント到達可能性 [1] を定義している．本論文ではアカウント到達可能性を求める具体的な手法のひとつとして，SNS の投稿内容に含まれる地域情報をもとにアカウント到達可能性算出モデルを検討する．

本論文は，2 節でアカウント到達可能性の定義とモデルの検討を行い， 3 節では地域情報に着目したアカウント到達可能性算出モデルの検討を行っている．4 節では地域情報に着目したアカウント到達可能性算出モデルの構築のための検証を 3 つ行っている．5 節はまとめであり， 6 節が関連研究となっている．

2. アカウント到達可能性算出モデルの検討

2.1 アカウント到達可能性

アカウント到達可能性 (Account Reachability) とは攻撃者が利用者の既知のアカウントから別のアカウントを見つけ出す可能性を表す．たとえば，ある利用者が二つの異なる SNS のアカウント s_1 ， s_2 をそれぞれ持っているとする．また攻撃者は利用者の SNS アカウントのうち， s_1 のみしか知らないとする．攻撃者は s_1 の情報をもとにして，まだ知らないアカウントである s_2 をさまざまな手法を通して見つけ出そうとする．ここで攻撃者は s_1 のプロファイルや投稿内容から s_1 のキーワードを抽出し，検索エンジンなどを用いて検索を行い，s_2 になりうるアカウントの候補を取得する手法をとったとする．このとき，取得した候補アカウントそれぞれと s_1 から取得した

(注1) : http://www.facebook.com
(注2) : https://twitter.com/

キーワードをもとに s_1 との類似度をはかり，s_2 が s_1 のアカウントであると特定していく．この可能性がアカウント到達可能性である．

アカウント到達可能性

アカウント s_1 から別のアカウント s_2 を見つけ出す可能性は以下に表される．

$$AR(s_1 \rightarrow s_2) = max_{q \subseteq\!\!\!= Q}\,(AR(s_1, s_2, q))$$

$$Q = GenQueries(s_1.prof, s_1.msg).$$

$$AR(s_1 \rightarrow s_2, q) =$$

$$Match(s_2, Cand(q)) * \frac{Score(s_1, s_2)}{\sum_{c \in Cand(q)} Score(s_1, c)}$$

$$Match(s_2) = \begin{cases} 1\ if\ s_2 \in Cand(q) \\ 0\ else \end{cases}.$$

ここで $GenQueries\ (s_1.prof, s_1.msg)$ は s_1 のプロファイルや投稿内容から，s_2 のアカウントを見つけ出すためのクエリを生成する式である．Q は生成されたクエリの集合であり，$Cand(q)$ はクエリ q で得られた s_1 の別アカウントの候補アカウントの集合である．$Score(s_1, c)$ は s_1 と候補アカウント c との類似度を表す．

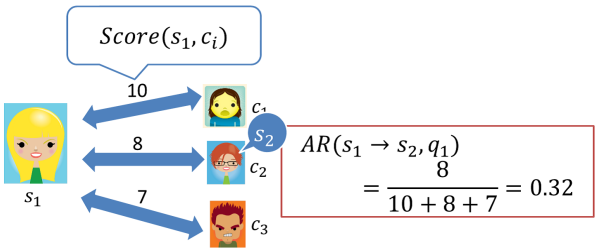

図 1　アカウント到達可能性算出イメージ

例をもとに，実際にアカウント到達可能性を求めてみる．攻撃者は s_1 のプロファイル情報をもとに，$Q = q_1, q_2$ のクエリを生成する．これらのクエリはそれぞれ，{”$keywords$”を”検索エンジン”を用いて検索する } といったクエリである．クエリ q_1 から c_1 ，c_2 ，c_3 の候補アカウントが取得でき，それぞれ s_1 との類似度を 10，8，7 とする．c_2 が s_2 であるとすると，$AR(s_1 \rightarrow s_2, q_1)$ は $8/(10+8+7) = 0.32$ となる．同様にクエリ q_2 から c_1 ... c_6 の候補アカウントが取得できたとする．それぞれ s_1 との類似度が 30，2，2，1，1，0 であり，c_1 が s_2 であるとき，$AR(s_1 \rightarrow s_2, q_2)$ は $30/(30+2+2+1+1+0) = 0.83$ となる．このように，それぞれのクエリで $AR(s_1 \rightarrow s_2, q)$ を求め，最終的に $AR(s_1 \rightarrow s_2)$ を求める．この例の場合，アカウント到達可能性は 2 つの値のうち大きい値である 0.83 となる．

アカウント到達可能性を求めるために用いられるこれらの関数は，攻撃者が入手できるデータや利用できる技術に基づいて実装される．先行研究 [1] では技術知識を持たない攻撃者でもできる最もシンプルな方法として，$GenQuery(s_1.prof, s_1.msg)$ では Profile に含まれる名前，所属をキーワードに検索エンジンで検索する実装を，$Score(s_1, c)$ では検索エンジンのランキングの逆数をとる類似度計算を採用した．

$GenQueries(s_1.prof, s_1.msg)$:
先行研究 [1] ではここで $KeywordSearch(ks, engine)$ という関数を導入した．検索エンジン $engine$ を用いてキーワード ks を検索する関数である．アカウント s_1 のプロファイルから名前，所属，誕生日といったキーワードを抽出する．それぞれのキーワードに対して，関数 $KeywordSearch(ks, engine)$ を適用する．この $KeywordSearch$ をクエリとする．

$Score(s_1, c)$:
検索エンジンを用いた検索の結果は，検索順位順に結果ページに出力される．これを用い $Score$ を以下のように定義した．

$$Score(s_1, c) = \frac{1}{Rank(ks, c)}$$

$Rank(ks, c) = \{ks$ を検索した結果ページにおける c の順位 $\}$

2.2　アカウント到達可能性算出モデルの検討

先行研究ではプロファイルデータのみを用いたもっともシンプルな攻撃モデルを採用したが，データ分析技術を用いることで，利用者の投稿履歴からも特徴的なキーワードを抽出し攻撃に利用することができる．$GenQueries(s_1.prof, s_1.msg)$ と $Score(s_1, c)$ を求める代表的な手法を以下の表に示す．

$GenQueries(s_1.prof, s_1.msg)$	$Score(s_1, c)$
プロファイルから生成	検索エンジンでのランキング
投稿内容から居住地を推測	検索エンジンでのランキング
投稿内容から特徴語を抽出	著者推定を利用し類似度を算出

表 1　$GenQueries(s_1.prof, s_1.msg)$ と $Score(s_1, c)$ の代表例

表に示す手法以外にも，さまざまな手法が考えられる．たとえば，友人関係を利用した方法も考えられる．文献 [3] では，異なるソーシャルグラフ間で同一人物であるノードを推定する手法を提案している．同一人物のノードであるとわかっている 2 つのソーシャルグラフの構造をトレーニングデータとして学習を行い，未知のノード間の類似性の推定を行っていく．

また投稿されるメッセージや写真にジオタグを付けている利用者も多くいる．これらのジオタグ情報をもとに，利用者の居住地などを推定する手法も考えられる．

先行研究や表 1 に示している手法において $Score$ の値は検索エンジンでの検索順位をもとに求めているが，より精度の高い類似度を測るためには候補アカウントの属性を抽出し，対象アカウントの属性との類似度を測ることが考えられる．代表 SNS である Twitter や Flickr のプロファイル記入欄は自由記述形式であり，機械的に利用者のプロファイル情報を抽出することは困難である．文献 [4] [5] では，自由記述形式のプロファイル欄や投稿内容，また友人関係から利用者の属性を抽出する

技術を提案している．

また著者推定を用いて類似度を測る手法も提案されている [6]．この手法は利用者が投稿した文章を元に，利用者の「癖」をテキストマイニングの技術を用いて見つけ類似度を測る手法である．

算出モデルにこれらの技術を実装することで，利用者が攻撃される可能性のあるさまざまな攻撃をシミュレートすることができるようになる．

3. 地域情報を利用したアカウント到達可能性算出モデル

3.1 定　　義

本論文ではその 1 手法として，SNS の投稿内容に含まれる地域情報をもとにした攻撃モデルを求め，それを用いてアカウント到達性を定義し，検証を行う．二つの関数の $GenQueries(s_1.prof, s_1.msg)$ と $Score(s_1, c)$ 実装を以下のようにする．

$GenQueries(s_1.prof, s_1.msg)$：
SNS1 の投稿内容から地域名を取得しキーワードとする．取得したキーワードを検索エンジンを用いて検索し，別アカウントの候補を見つけ出す．

$Score(s_1, c)$：
s_1 から得られた地域情報を $locate(s_1)$，候補者 c_i から得られた地域情報のうち $locate(s_1)$ と同じ地域情報を $locate(c_i)$ とする．各候補者の類似度を

$$Sim(s_1, c_i) = \frac{|locate(c_i)|}{|locate(s_1)|}$$

とする．これをもとに各候補者に対して仮の $Score$ を算出する．これを $PreScore(s_1, c)$ とすると

$$PreScore(s_1, c_i) = \frac{Sim(s_1, c_i)}{\sum_{c \in Cand(q)} Sim(s_1, c)}$$

として求める． $PreScore$ が 0.1 以上の候補アカウントのみを取り出し， $PreScore$ の値が大きい順にランキングしなおす．このランキングをもとにして，先行研究と同じ手法であるランキングの逆数をとったものを $Score(s_1, c)$ とする．

3.2 算　出　例

対象となる SNS を Twitter と Facebook に仮定する．このとき $GenQueries$ と $Score$ の算出例を示す．

例 1 Twitter → Facebook
既知のアカウントが Twitter であり，見つけ出すアカウントを Facebook とする．

$GenQueries$：
Twitter の投稿内容（Tweet）から地域情報を抽出する．Tweet 内に出現する頻度の高い地域名をキーワードとし，*Google* などの検索エンジンや Facebook の友人検索機能を用いて Facebook アカウントの検索を行う．

$Score$：
検索の結果得られた Facebook アカウントから地域名を抽出し，得られた地域名と Twitter から抽出していた地域名を用いて $Score$ を求めていく．

例 2 Facebook → Twitter
既知のアカウントが Facebook であり，見つけ出すアカウントを Twitter とする．

$GenQueries$：
Facebook の投稿内容から地域情報を抽出する．投稿内容内に出現する頻度の高い地域名をキーワードとし，*Google* などの検索エンジンを用いて Twitter アカウントの検索を行う．

$Score$：
検索の結果得られた Twitter アカウントから地域名を抽出し，得られた地域名と Facebook から抽出していた地域名を用いて $Score$ を求めていく．

4. 検　　証

SNS から抽出した地域情報が，本当に別アカウントを見つけるキーワードになり得るのか検証するために 3 つの検証を行った．ここで 3.2 節で挙げた例をもとに，対象 SNS を Twitter と Facebook とする．

検証 1：　Twitter と Facebook の両方のアカウントを持つ利用者に対し， Tweet と Facebook の投稿内容から地域名を抽出し，どれだけ類似しているか

検証 2：　Tweet から抽出した地域名で，利用者の別アカウントが取得できるか

検証 3：　Facebook のプロファイルから抽出した地域名で，利用者の別アカウントが取得できるか

4.1 検　　証 1

複数の SNS の利用者は各 SNS の利用方法をもとに，大きく以下の 4 タイプに分類することができる．

（1） 複数の SNS アカウントが同一人物と知られても構わない利用者で，投稿内容も類似している
（2） 複数の SNS アカウントが同一人物と知られても構わない利用者だが，投稿内容は類似していない
（3） 複数の SNS アカウントが同一人物と知られたくない利用者だが，投稿内容が類似している
（4） 複数の SNS アカウントが同一人物と知られたくない利用者で，投稿内容も類似していない

タイプ 3 の利用者は，使い分けをしているつもりでできていない利用者である．我々が注目すべき利用者はタイプ 3 の利用者であり，これらの利用者がサイバーストーカーの被害にあっ

た場合多くの被害をこうむることになる．

本予備実験では，Tweet などに含まれる地域名に着目し被験者を以下の 4 タイプに分類する．

（1） 複数の SNS アカウントが同一人物と知られても構わない利用者で，投稿内容に含まれる地域名も類似している
（2） 複数の SNS アカウントが同一人物と知られても構わない利用者だが，投稿内容に含まれる地域名は類似していない
（3） 複数の SNS アカウントが同一人物と知られたくない利用者だが，投稿内容に含まれる地域名が類似している
（4） 複数の SNS アカウントが同一人物と知られたくない利用者で，投稿内容に含まれる地域名も類似していない

被験者は Twitter と Facebook の両方のアカウントを持つ 50 名であり，26 名は両方のアカウントが第三者に同一人物だと知られても良いと考えている被験者である．24 名の被験者は両方のアカウントが第三者に同一人物だと知られたくない被験者である．各被験者に対し，tweet と Facebook の投稿内容また Facebook のプロファイルから地域名を取得し，取得した地域名がいくつ重複しているのか確認した．

取得した Tweet 数は各被験者 1000 件， Facebook の投稿数は約 100 件であり， Facebook のプロファイルから地域名も取得する．取得した Tweet または Facebook の投稿文章に対しストップワードの除去を行い，形態素解析器 MeCab [10] を用いて形態素解析を行った．形態素解析の結果，地域名のみを抽出し，検証を行った．

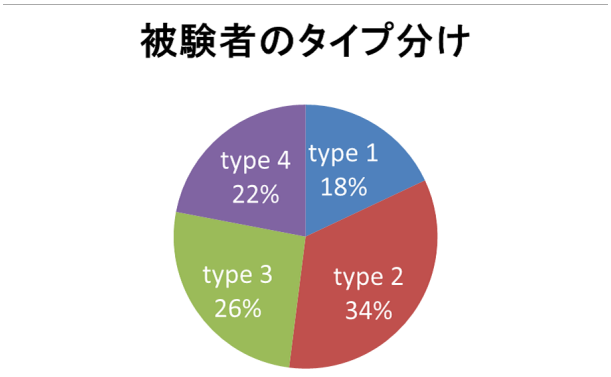

図 2 被験者のタイプ分け

結果を図 2 に示す．被験者のうち 26% がタイプ 3 の「複数の SNS アカウントが同一人物と知られたくない利用者だが，投稿内容に含まれる地域名が類似している」利用者であることが分かった．タイプ 3 に分類された被験者を対象に Tweet から抽出した地域名上位 10 件が被験者にとってどのような地域なのか，またどのような地域名が Facebook から抽出した地域名と類似しているのか検証を行った．タイプ 3 の被験者から代表 4 名の結果を図 2 に示す．

赤い太文字で書かれている地域は， Facebook から抽出した地域と同一の地域である．彼らの特徴として，居住地が Tweet 内に出現する地域名の中で最も出現しており，またそれらが Facebook 上でも取得可能であることがいえる．被験者にヒアリングを行ったところ， Twitter を利用しているうちにいつの間にかつぶやいてしまっていたり，友人とのやり取りをするうちに地域名をつぶやいてしまっているとの回答を得た．居住地は SNS 利用者にとっても漏えいしては困る個人情報のひとつであり，地域名を利用してアカウント到達可能性を求める手法は有用であるといえる．

4.2 検　証 2

検証 1 では Tweet から抽出した地域名と Facebook から抽出した地域名の類似を見た．検証 2 では 3.2 節の例 1 をもとに，Tweet から抽出した地域名をキーワードとしたとき，被験者の別のアカウントを見つけることができるか検証を行った．

被験者は検証 1 で被験者となった SNS 利用者 50 名である．キーワードは Tweet から抽出した地域名上位 5 件であり，各キーワードを組み合わせてクエリの生成を行う．検索には Facebook の友人検索と *Google* 検索エンジンを利用した．

被験者 50 名全員， Facebook の友人検索と *Google* 検索エンジン両方の検索結果に現れなかった．Facebook の友人検索ではスポット検索も同時に行っており，地域名一つで検索を行うと個人ページの検索は行われず，目的の利用者の検索はできない．また，二つ以上の地域名を組み合わせて検索を行うと，どれか一つの地域名を人名と判断して検索が行われてしまう．これは Facebook 上で全く知らない利用者を安易に見つけることができないようになっていることが考えられる．*Google* などの検索エンジンを用いる場合でも，個人のページは安易に検索されないようになっている．Facebook は多くの個人情報を掲載する SNS である．そのため，安易な検索によって個人情報が漏れることを防ぐために Facebook 側で検索を制限していることが考えられる．

この調査から，Twitter から抽出した地域情報だけを用いて別のアカウントを見つけ出すことは難しく，利用者にとって少しではあるが安心できる結果が得られた．

4.3 検　証 3

検証 2 によって， Twitter から抽出した地域情報だけを用いて別アカウントを見つけ出すことは難しいことが分かった．検証 3 では Facebook をもとに Twitter のアカウントを見つけ出すことを考える．先行研究 [1] では Facebook のプロファイル情報からキーワードを生成し，それを元に Twitter アカウントを見つけ出す実験を行っている．ここでは， *Score* を 3.1 節で定義したものを用いて再度アカウント到達可能性を算出し，先行研究で得られたアカウント到達可能性と比べ，考察を行う．

被験者は 10 名であり，それぞれ先行研究においてアカウント到達可能性が求められている被験者である．ここで $GenQueries(s_1.prof, s_1.msg)$ は先行研究で利用した，プロファイル情報をもとにキーワードを生成する手法を用いる．

Score で類似度を求める際に利用する地域情報は Facebook においてはプロファイルから取得する．利用者が登録している出身地，居住地，また情報を多くするために出身学校の所在地を取得している．Twitter においてはプロファイル欄と Tweet からそれぞれ地域名を抽出している．検索には *Google* 検索

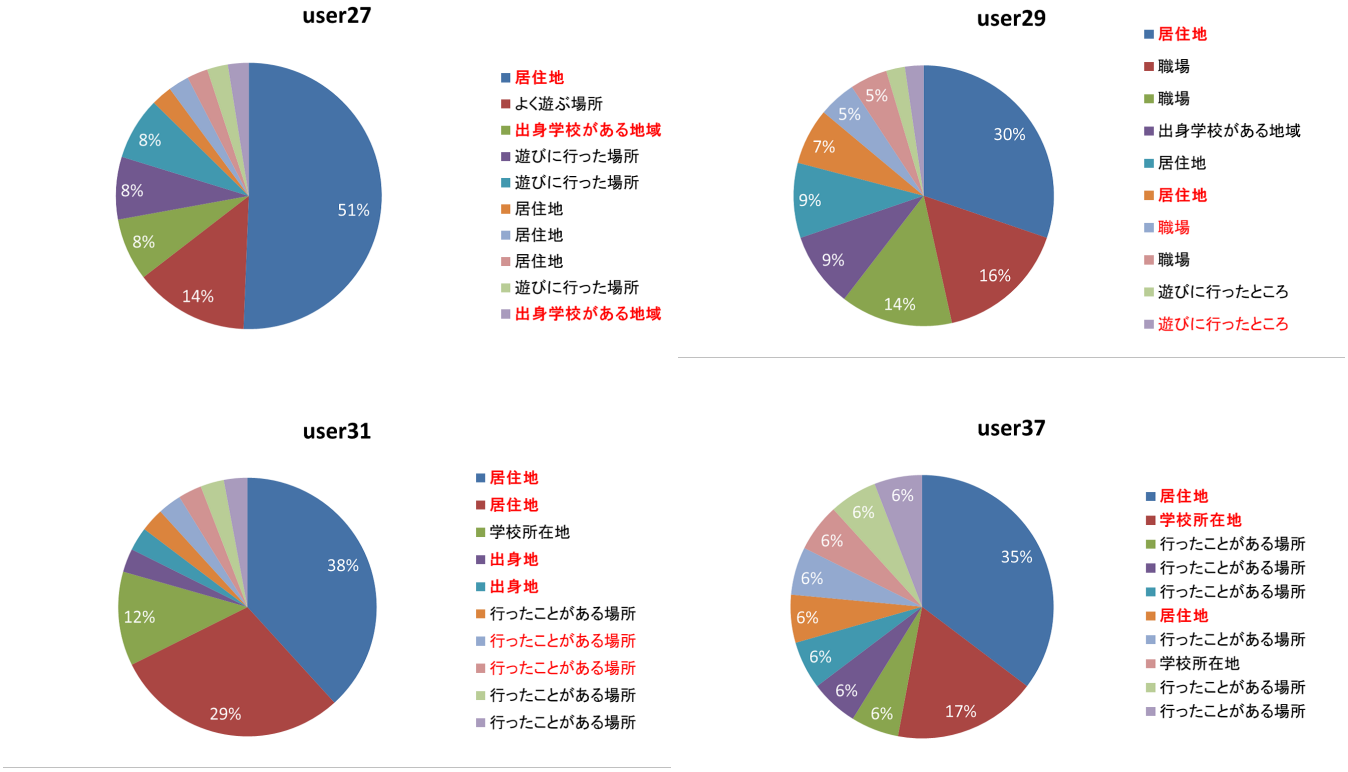

図 3　タイプ 3 の地域名分類

エンジンを利用した．Twitter アカウントを見つけ出すため，*site* : *twitter.com* をキーワードの最後に加えている．

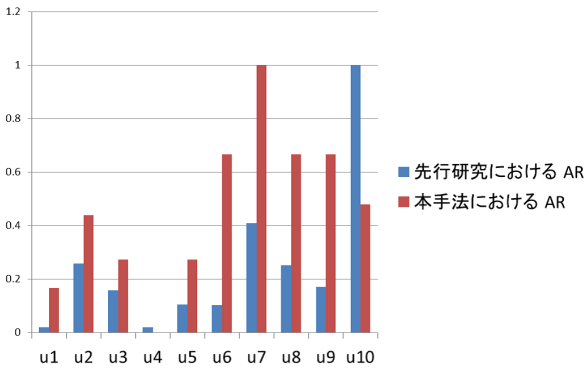

図 4　検証 3 結果 (アカウント到達可能性)

検証結果を図 4 に示す．全被験者のうち 8 名の被験者が，先行研究で得られたアカウント到達可能性よりも高いアカウント到達可能性の値が得られた．これは，候補アカウントそれぞれの Tweet の中身までみて類似度を測るため，より被験者に近いアカウントが抽出されたことが考えられる．先行研究よりも低いアカウント到達可能性が得られた被験者 2 名は投稿内容に自身のプロファイルに近い地域情報を含んでいないことが分かった．

各被験者で得られた候補アカウント数は最終的に少ないもので 1 件，多いものでも 11 件にしぼられた．

図 5 ，図 6 に被験者代表 1 名の *PreScore* と *Score* を示す．*Sim* は 4 つの条件に分けて算出した．Facebook のプロ

Cand	a_I	a_{II}	b_I	b_{II}	PreScore
c1	0	0	0.333333	0.333333	0.25
c2	0	0	0	0	0
c3	0	0	0	0	0
c4	0	0	0.166667	0	0.0625
c5	0	0	0	0	0
c6	0	0.5	0	0.166667	0.25
c7	0	0	0	0	0
c8	0	0	0	0.166667	0.0625
c9	0	0	0	0.166667	0.0625
c10	0	0.5	0	0.333333	0.3125

図 5　検証 3 結果 (*PreScore*)

Cand	Rank	Score
c1	2	0.5
c6	3	0.33
c10	1	1

図 6　検証 3 結果 (*Score*)

ファイルから得られる地域情報を，出身地と居住地から得られた地域情報 (a) と，出身学校の所在地を含めた地域情報 (b) に，Twitter の候補アカウントから得られる地域情報を，プロファイル欄から得られた地域情報 (I) と，Tweet から得られた地域情報 (II) に分け，これらを組み合わせて条件を設定する．

もともと得られた候補アカウント数は 10 件であったが，$PreScore$ 算出後 3 件に絞ることができた．c_1 が被験者のアカウントである． c_6 のアカウントが被験者のアカウントと似ていることがわかる．しかし被験者と c_6 は全く異なる利用者であり，Sim の値も各条件によって異なる．地域情報だけでは個人を特定することは難しい一方で，類似している利用者が意外と少ないこともわかる．今回はすべての条件を均等に評価しているため，今後はそれぞれの条件に比重をおいて評価を行っていきたい．

5. 関 連 研 究

SNS の投稿内容を用いた居住地推定の手法は多く提案されている．文献 [2] では， Twitter の投稿内容のみを用いて利用者の居住地を推定するモデルを提案している．市町村レベルの居住地の推定を目標としている．まずそれぞれの市町村に対し，各市町村に関連する Tweet 内に出現しやすい地域特有の単語 (local word) を設定する．local word ひとつひとつには各市町村での自身の出現確率が与えられている．利用者の居住地推定を行うために，Tweet に出現する local word を抽出し， local word の出現頻度をもとに，各地域それぞれに対し居住している可能性を求めていく．最も高い可能性を持つ地域を利用者の居住地と推定する．

今後これらの技術も取り入れ，よりよいモデルの構築を行っていきたい．

6. まとめと今後の課題

近年 SNS の普及とともに，SNS の利用において自身が投稿した内容やプロファイルがどの程度のプライバシリスクになっているかを把握する必要性が増している．サイバーストーカーなどの被害に遭うリスクは SNS の利用者全員が持っており，どのように被害を防ぐかが重要になる．我々は，SNS におけるプライバシリスクの提示指標として，アカウント到達可能性を定義した．本論文ではアカウント到達可能性を算出するモデルの一検討として，居住地推測を用いたモデルの検討を行った．

Twitter から地域名を取得し， Facebook の投稿内容やプロファイルから抽出できる地域名との類似度をはかることでアカウント到達可能性を求めることを目標とする．本論文では，Twitter から抽出した地域名が実際に利用者の個人情報に近いものであるのか，また Facebook から抽出できる地域名との類似性はあるのかを検証するため予備実験を行った．

実験の結果，Twitter から抽出した地域名は実際位に利用者の個人情報に近いものが多く抽出され，また Facebook から抽出できる地域名との類似性もみられた．しかしながら，Facebook の友人検索機能を用いて Twitter から抽出した地域名を検索したところ，対象の被験者を見つけることができなかった．Facebook では安易にユーザ検索されないために，人名が含まれないキーワードの組合せの場合でも，どれか一つを人名として検索を行ってしまう．また検索エンジンを利用して検索する場合，個人のページは人名がクエリに入っていない場合検索できないようになっている．このように Facebook は個人ページが検索されづらいように制限されており，利用者にとって少しではあるが安心できる結果が得られた．

しかしながら，Facebook から得られた地域情報をキーワードとして Twitter アカウントを探すと簡単に見つかってしまう可能性がある．今後，どのような情報を SNS 上で公開することで，個人が特定される危険が高まるか調査していきたい．

我々の目標は，利用者自身が SNS の利用において自身が投稿した内容やプロファイルがどの程度のプライバシリスクになっているかを把握することである．今後の課題として，よりサイバーストーカーが実際に個人情報を取得していく過程に近い手法でプライバシリスクの把握ができるようにすることがあげられる．サイバーストーカーは SNS の文章から利用者がどの地域にいるのか推測することができる．彼らは地域特有の単語や言い回しから推測していると考えられる．そこで Twitter から抽出する地域情報を地域名だけでなく，各地域特有の単語を利用するなどさまざまな手法で地域情報を抽出することを考えていくことがあげられる．たとえば,「東京浅草」 という地域特有の単語として 「浅草寺」 や 「雷門」 などが考えられる．このように地域特有の単語を利用することで，より多くの地域情報の抽出が可能である．

また，本論文では地域情報のみを用いたアカウント到達可能性算出モデルの検討を行ったが，アカウント到達可能性を算出する手法はさまざまある．今後これらの検討も行っていきたい．

文　献

[1] Ayano YOSHIKUNI, and Chiemi WATANABE ”Account Reachability : A Measure of Privacy Risk for Exposure of a User’s Multiple SNS Accounts”, Proceedings of the 15th International Conference on Information Integration and Web-based Applications & Services (iiWAS2013)

[2] Zhiyuan Cheng, James Caverlee, and Kyumin Lee ”You Are Where You Tweet: A Content-Based Approach to Geo-locating Twitter Users”, CIKM’ 10

[3] Narayanan Arvind and Shmatikov Vitaly ”De-anonymizing Social Networks”, Proceedings of the 2009 30th IEEE Symposium on Security and Privacy

[4] Rao Delip and Yarowsky David and Shreevats Abhishek and Gupta Manaswi ”Classifying latent user attributes in twitter”, SMUC ’10

[5] Mislove Alan and Viswanath Bimal and Gummadi Krishna P. and Druschel Peter ”You are who you know: inferring user profiles in online social networks”, WSDM ’10

[6] Stamatatos Efstathios ”A survey of modern authorship attribution methods”, J. Am. Soc. Inf. Sci. Technol., 60(3):538-556, mar 2009

[7] アウンコンサルティング株式会社,「世界 40ヶ国のフェイスブック（facebook）人口推移【2013 年 5 月】」, http://www.auncon.co.jp/corporate/2013/0605.html
[8] BUSINESS INSIDER, "Twitter Is Surprisingly Small Compared To A Bunch Of Other Apps And Online Companies", http://www.businessinsider.com/twitter-user-base-compared-to-other-apps-and-online-companies-2013-11
[9] 「【Twitter 実験】つぶやきだけで個人を特定できるのか？」, http://picup.omocoro.jp/?eid=1315
[10] MeCab: Yet Another Part-of-Speech and Morphological Analyzer, http://mecab.googlecode.com/svn/trunk/mecab/doc/index.html